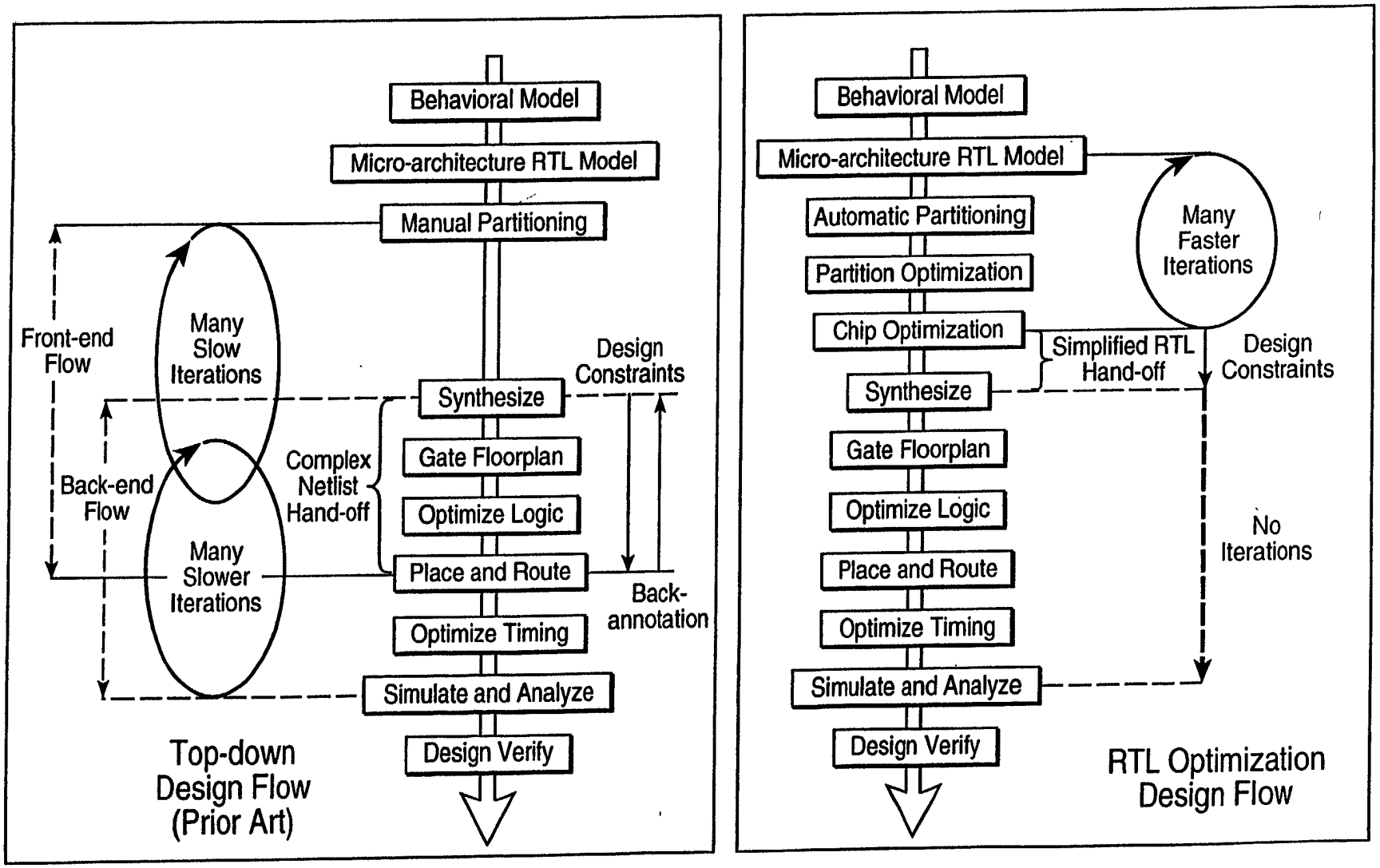


FIG. 1

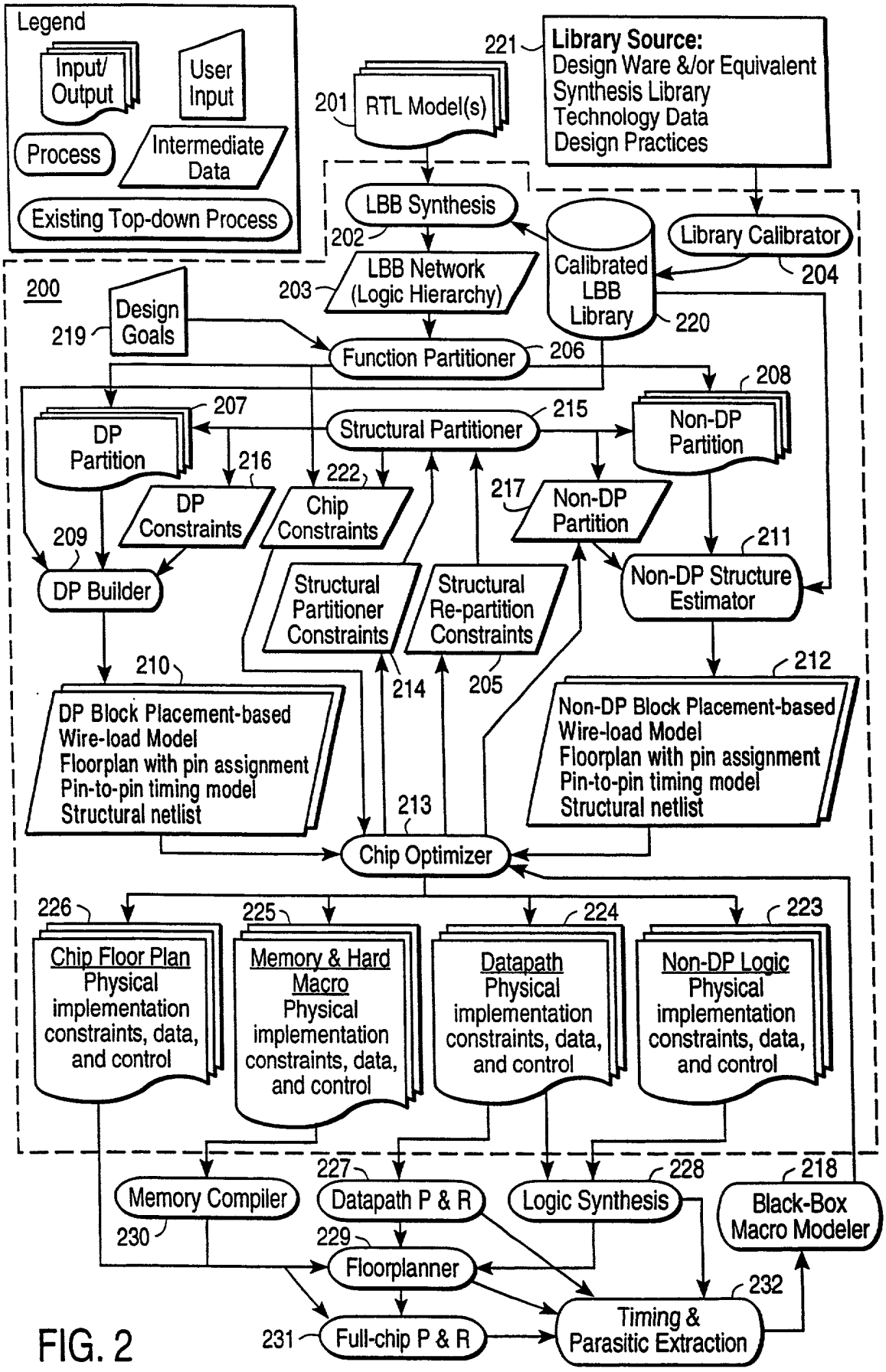

Legend
Input/ Output
User Input
Process
Intermediate Data
Existing Top-down Process
221
Library Source:
Design Ware &/or Equivalent
Synthesis Library
Technology Data
Design Practices
201
RTL Model(s)
LBB Synthesis
202
Library Calibrator
204
Calibrated LBB Library
220
200
219
Design Goals
203
LBB Network (Logic Hierarchy)
Function Partitioner
206
208
207
DP Partition
Structural Partitioner
215
Non-DP Partition
216
222
DP Constraints
Chip Constraints
217
Non-DP Partition
209
211
DP Builder
Non-DP Structure Estimator
Structural Partitioner Constraints
Structural Re-partition Constraints
210
214
205
212
DP Block Placement-based
Wire-load Model
Floorplan with pin assignment
Pin-to-pin timing model
Structural netlist
Non-DP Block Placement-based
Wire-load Model
Floorplan with pin assignment
Pin-to-pin timing model
Structural netlist
213
Chip Optimizer
226
225
224
223
Chip Floor Plan
Physical implementation constraints, data, and control
Memory & Hard Macro
Physical implementation constraints, data, and control
Datapath
Physical implementation constraints, data, and control
Non-DP Logic
Physical implementation constraints, data, and control
227
228
218
Memory Compiler
Datapath P & R
Logic Synthesis
Black-Box Macro Modeler
230
229
Floorplanner
232
Timing & Parasitic Extraction
231
Full-chip P & R

FIG. 2

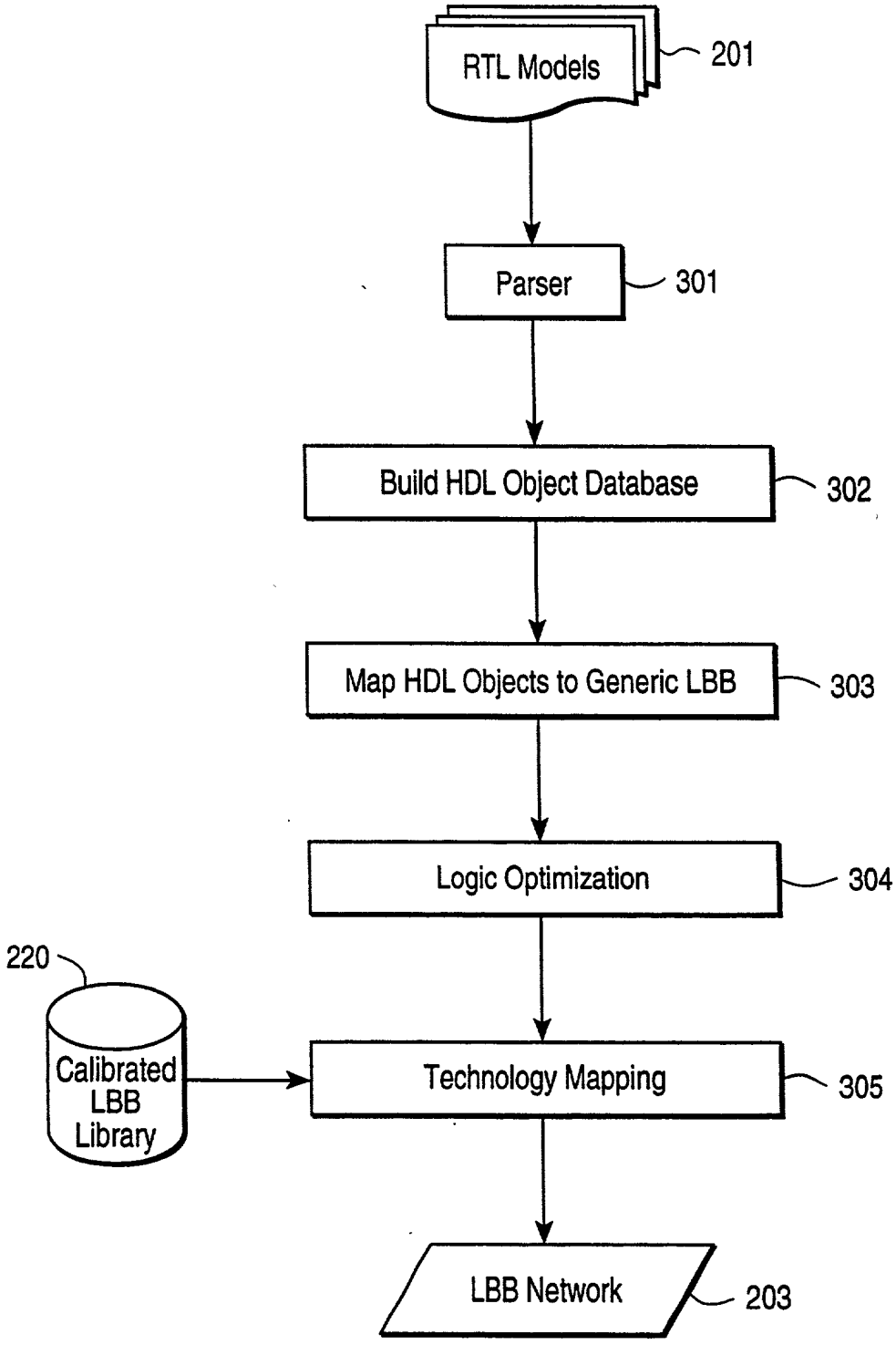

RTL Models
201
Parser
301
Build HDL Object Database
302
Map HDL Objects to Generic LBB
303
Logic Optimization
304
220
Calibrated LBB Library
Technology Mapping
305
LBB Network
203

FIG. 3

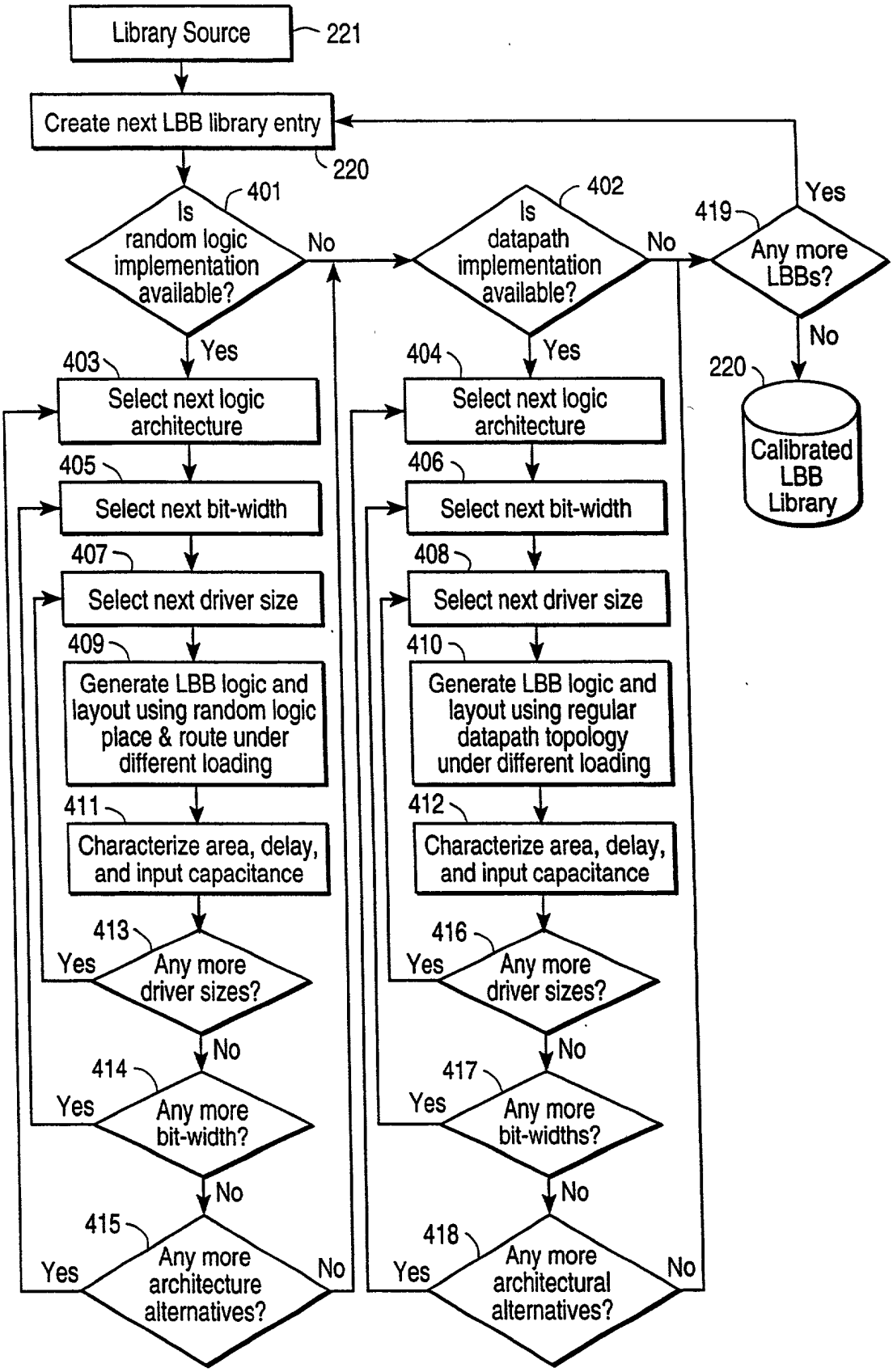

Library Source
221
Create next LBB library entry
220
401
Is random logic implementation available?
No
Yes
402
Is datapath implementation available?
No
Yes
419
Any more LBBs?
Yes
No
220
Calibrated LBB Library
403
Select next logic architecture
404
Select next logic architecture
405
Select next bit-width
406
Select next bit-width
407
Select next driver size
408
Select next driver size
409
Generate LBB logic and layout using random logic place & route under different loading
410
Generate LBB logic and layout using regular datapath topology under different loading
411
Characterize area, delay, and input capacitance
412
Characterize area, delay, and input capacitance
413
Any more driver sizes?
Yes
No
416
Any more driver sizes?
Yes
No
414
Any more bit-width?
Yes
No
417
Any more bit-widths?
Yes
No
415
Any more architecture alternatives?
Yes
No
418
Any more architectural alternatives?
Yes
No

FIG. 4

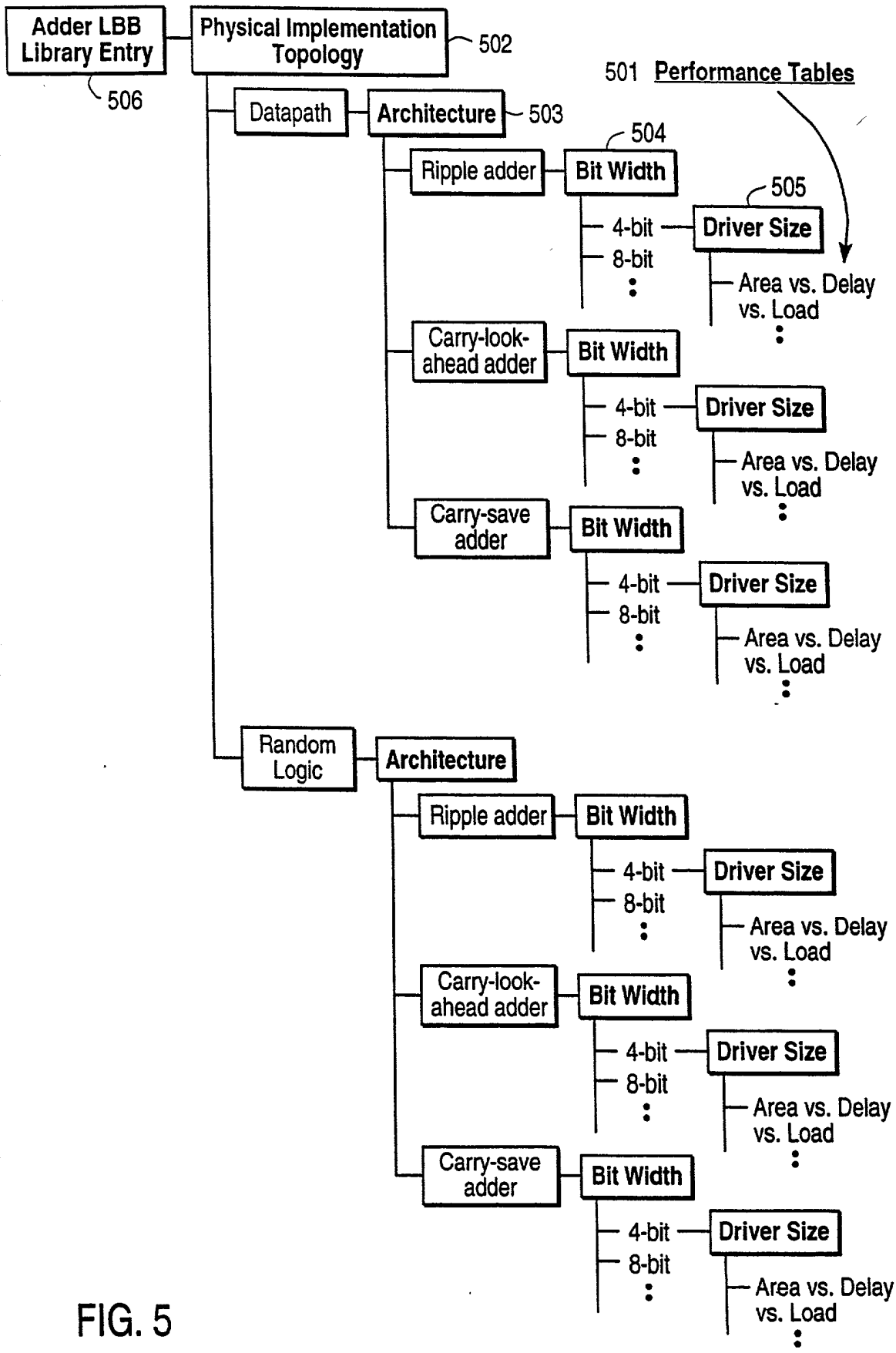

Adder LBB Library Entry
506
Physical Implementation Topology
502
501 Performance Tables
Datapath
Architecture
503
504
Ripple adder
Bit Width
505
4-bit
8-bit
Driver Size
Area vs. Delay vs. Load
Carry-look-ahead adder
Bit Width
4-bit
8-bit
Driver Size
Area vs. Delay vs. Load
Carry-save adder
Bit Width
4-bit
8-bit
Driver Size
Area vs. Delay vs. Load
Random Logic
Architecture
Ripple adder
Bit Width
4-bit
8-bit
Driver Size
Area vs. Delay vs. Load
Carry-look-ahead adder
Bit Width
4-bit
8-bit
Driver Size
Area vs. Delay vs. Load
Carry-save adder
Bit Width
4-bit
8-bit
Driver Size
Area vs. Delay vs. Load

FIG. 5

Category	LBB
Adder	Adder, AdderCi, AdderCiCo, AdderCo
Arithmetic	ALU, Modulus, Multiplier, Divider, UnaryMinus
Buffers	Buffer, TriStateBuf
Comparator	Equal, Greater, GreaterOrEqual, Less, LessOrEqual, NotEqual
Complex Logic	Finite State Machine, Hard Macros
Decoder	BinaryDecoder
Decrementer	Decrementer, DecrementerCi, DecrementerCiCo, DecrementerCo
Encoder	BinaryEncoder, PriorityEncoder
Incrementer	Incrementer, IncrementerCi, IncrementerCiCo, IncrementerCo
Logic Gates	And, Nand, Nor, Or, Xnor, Xor, Inverter, And-Or (AO), And-Or-Invert (AOI)
Memory	RAM, ROM, Register File
Mux	Mux, PriorityMux, UnencodedMux (AOI), InvMux, InvPriorityMux, InvUnencodedMux
Reduce Gates	ReduceAnd, ReduceNand, ReduceNor, ReduceOr, ReduceXnor, ReduceXor
Shift	ArithmeticShift, CircularLeftShift, CircularRightShift, ShiftLeft, ShiftRight
Storage	Latch, FlipFlop
Subtractor	Subtractor, SubtractorCi, SubtractorCiCo, SubtractorCo

FIG. 6

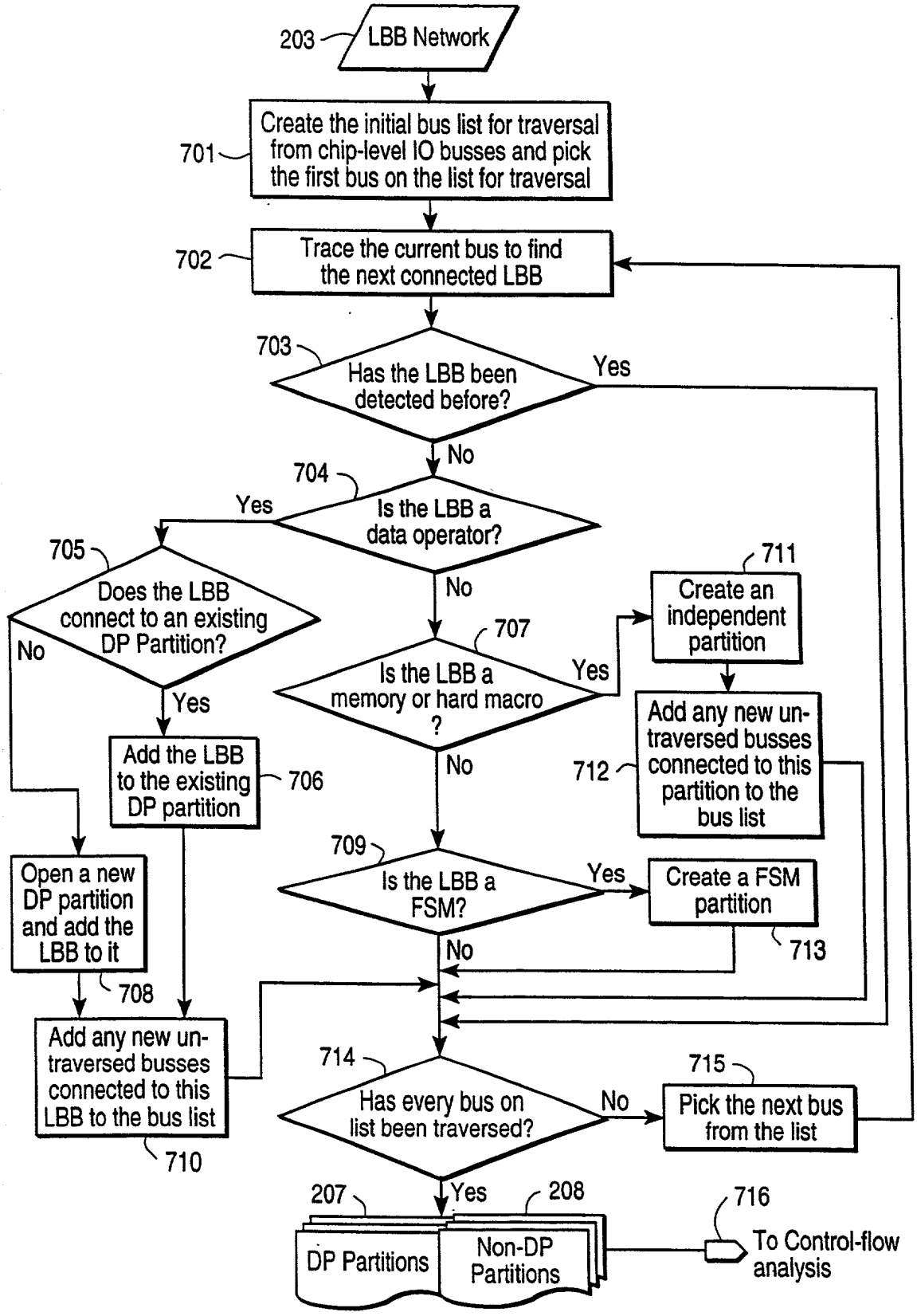

203
LBB Network
701
Create the initial bus list for traversal from chip-level IO busses and pick the first bus on the list for traversal
702
Trace the current bus to find the next connected LBB
703
Has the LBB been detected before?
Yes
No
704
Is the LBB a data operator?
Yes
No
705
Does the LBB connect to an existing DP Partition?
No
Yes
Add the LBB to the existing DP partition
706
Open a new DP partition and add the LBB to it
708
707
Is the LBB a memory or hard macro ?
Yes
No
711
Create an independent partition
712
Add any new un-traversed busses connected to this partition to the bus list
709
Is the LBB a FSM?
Yes
No
Create a FSM partition
713
Add any new un-traversed busses connected to this LBB to the bus list
710
714
Has every bus on list been traversed?
No
715
Pick the next bus from the list
Yes
207
DP Partitions
208
Non-DP Partitions
716
To Control-flow analysis

FIG. 7

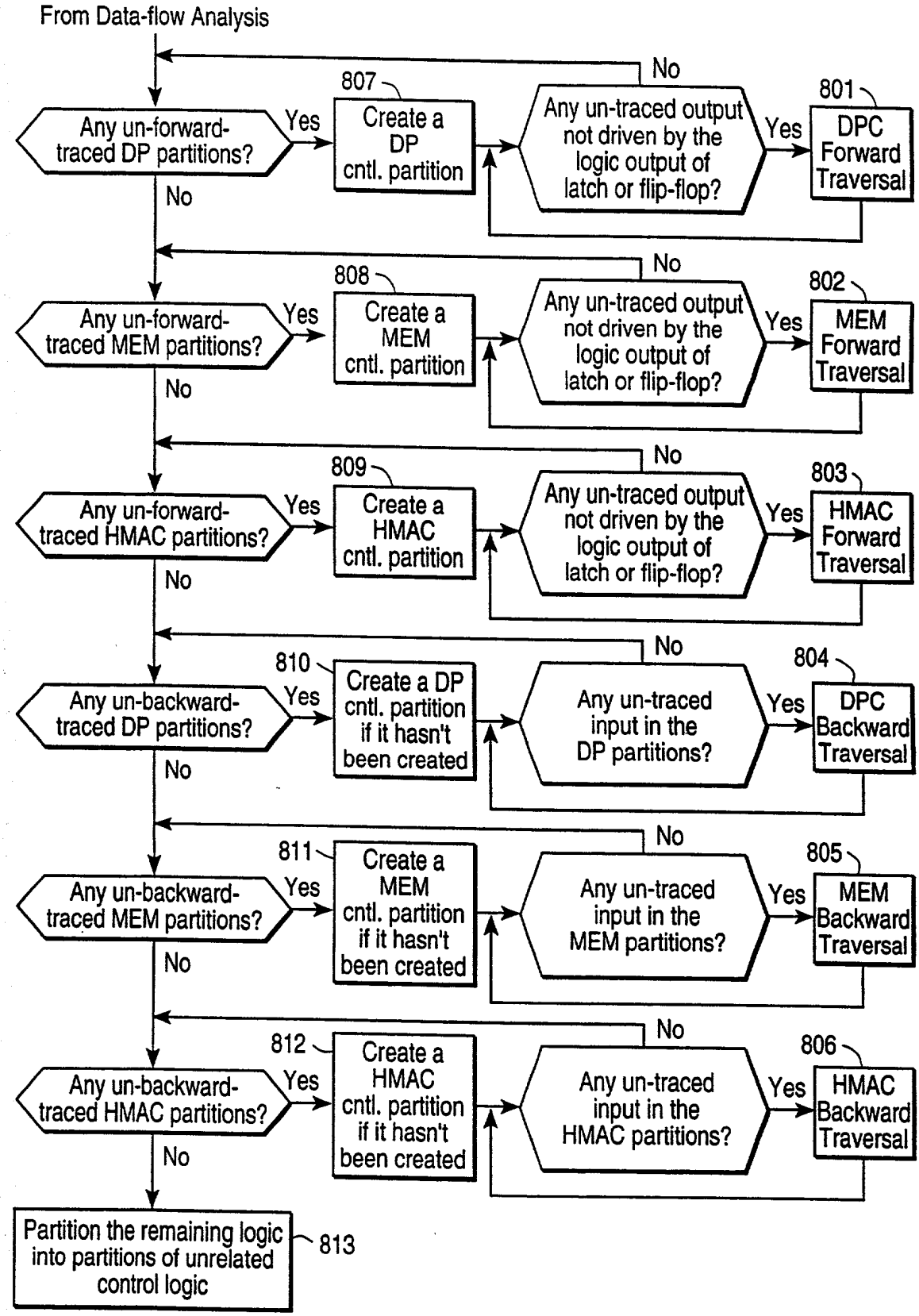

From Data-flow Analysis
Any un-forward-traced DP partitions?
Yes
No
807
Create a DP cntl. partition
Any un-traced output not driven by the logic output of latch or flip-flop?
Yes
No
801
DPC Forward Traversal
Any un-forward-traced MEM partitions?
Yes
No
808
Create a MEM cntl. partition
Any un-traced output not driven by the logic output of latch or flip-flop?
Yes
No
802
MEM Forward Traversal
Any un-forward-traced HMAC partitions?
Yes
No
809
Create a HMAC cntl. partition
Any un-traced output not driven by the logic output of latch or flip-flop?
Yes
No
803
HMAC Forward Traversal
Any un-backward-traced DP partitions?
Yes
No
810
Create a DP cntl. partition if it hasn't been created
Any un-traced input in the DP partitions?
Yes
No
804
DPC Backward Traversal
Any un-backward-traced MEM partitions?
Yes
No
811
Create a MEM cntl. partition if it hasn't been created
Any un-traced input in the MEM partitions?
Yes
No
805
MEM Backward Traversal
Any un-backward-traced HMAC partitions?
Yes
No
812
Create a HMAC cntl. partition if it hasn't been created
Any un-traced input in the HMAC partitions?
Yes
No
806
HMAC Backward Traversal
Partition the remaining logic into partitions of unrelated control logic
813

FIG. 8

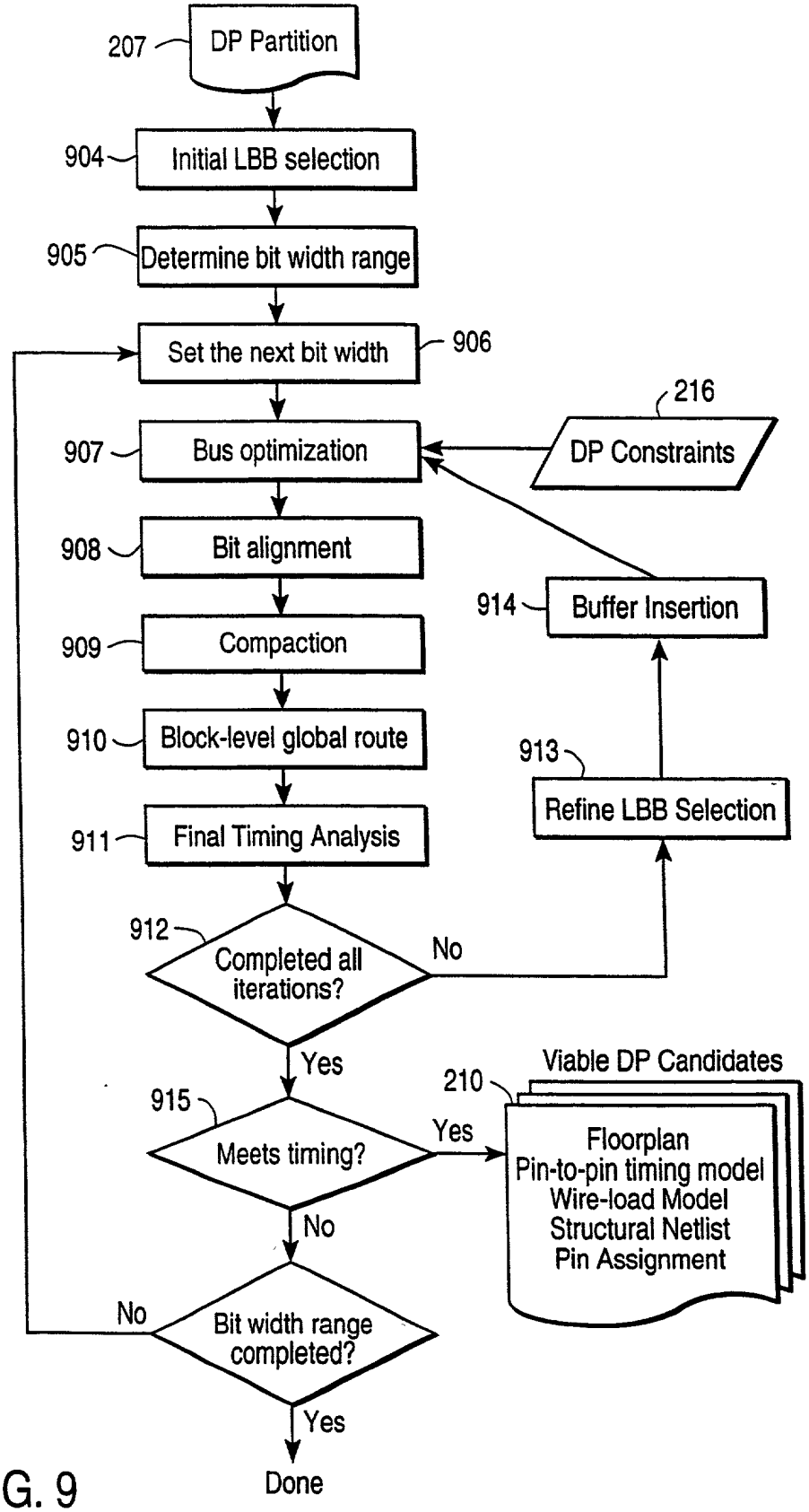
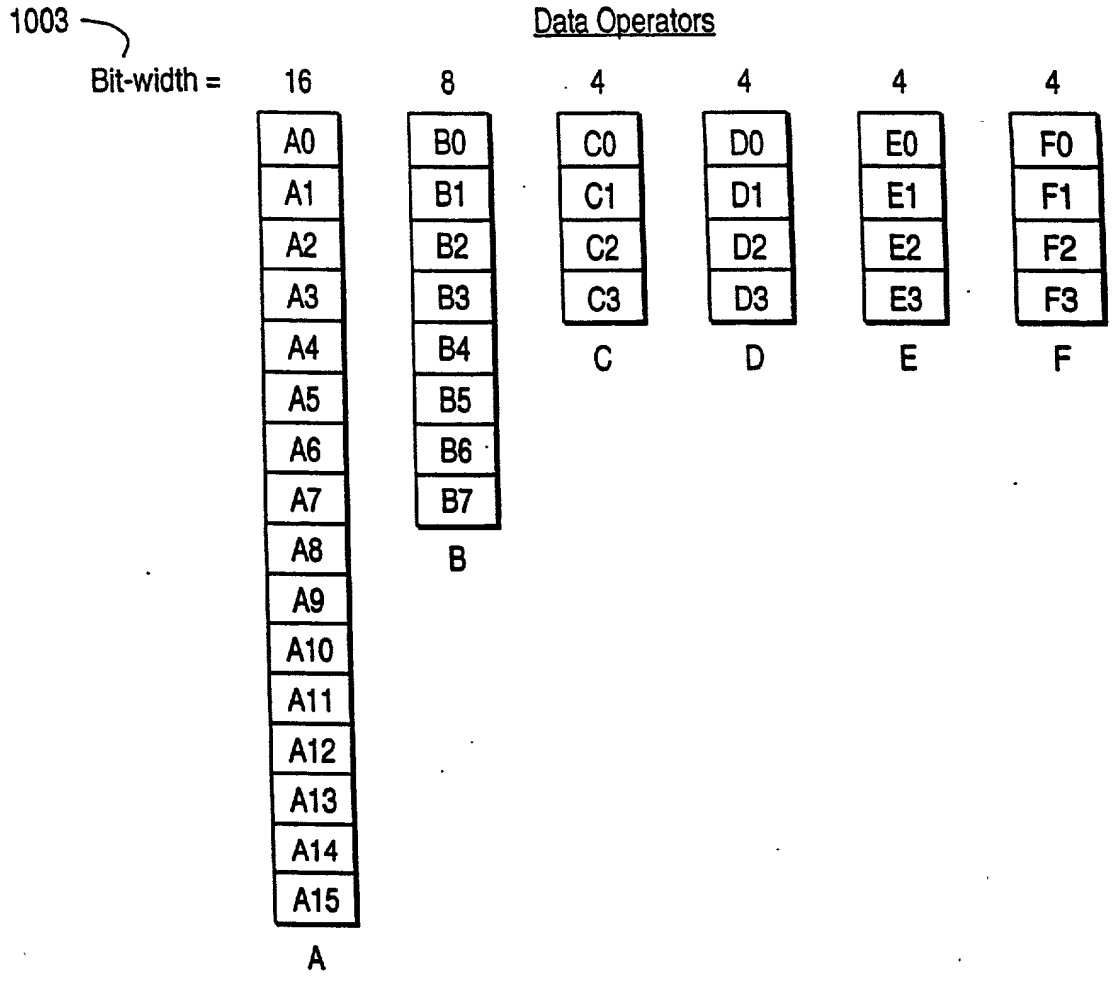

207
DP Partition
904
Initial LBB selection
905
Determine bit width range
906
Set the next bit width
216
907
Bus optimization
DP Constraints
908
Bit alignment
914
Buffer Insertion
909
Compaction
910
Block-level global route
913
Refine LBB Selection
911
Final Timing Analysis
912
Completed all iterations?
No
Yes
Viable DP Candidates
210
915
Meets timing?
Yes
Floorplan
Pin-to-pin timing model
Wire-load Model
Structural Netlist
Pin Assignment
No
No
Bit width range completed?
Yes
Done

FIG. 9


1003
Data Operators
Bit-width =
16
8
4
4
4
4
A0
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
A11
A12
A13
A14
A15
A
B0
B1
B2
B3
B4
B5
B6
B7
B
C0
C1
C2
C3
C
D0
D1
D2
D3
D
E0
E1
E2
E3
E
F0
F1
F2
F3
F

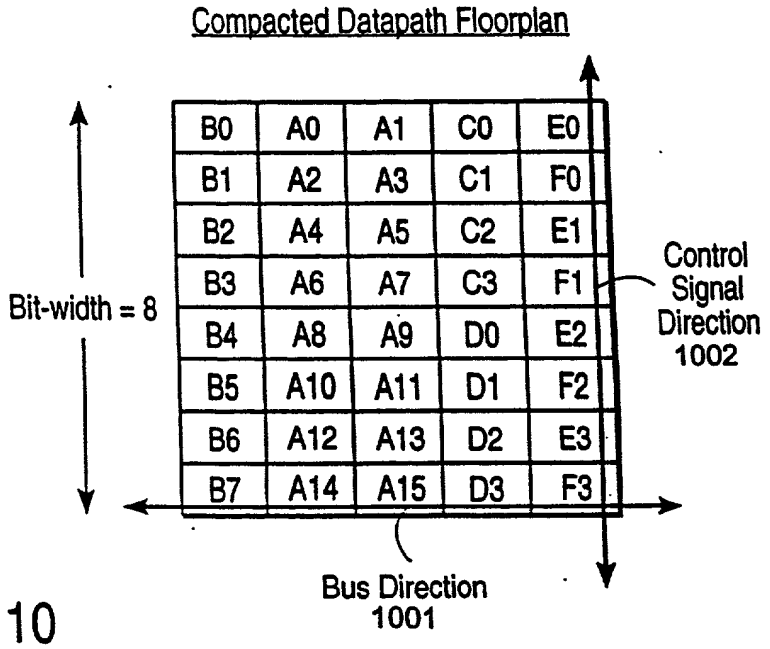

Compacted Datapath Floorplan
B0 A0 A1 C0 E0
B1 A2 A3 C1 F0
B2 A4 A5 C2 E1
B3 A6 A7 C3 F1
B4 A8 A9 D0 E2
B5 A10 A11 D1 F2
B6 A12 A13 D2 E3
B7 A14 A15 D3 F3
Bit-width = 8
Control Signal Direction 1002
Bus Direction 1001

FIG. 10

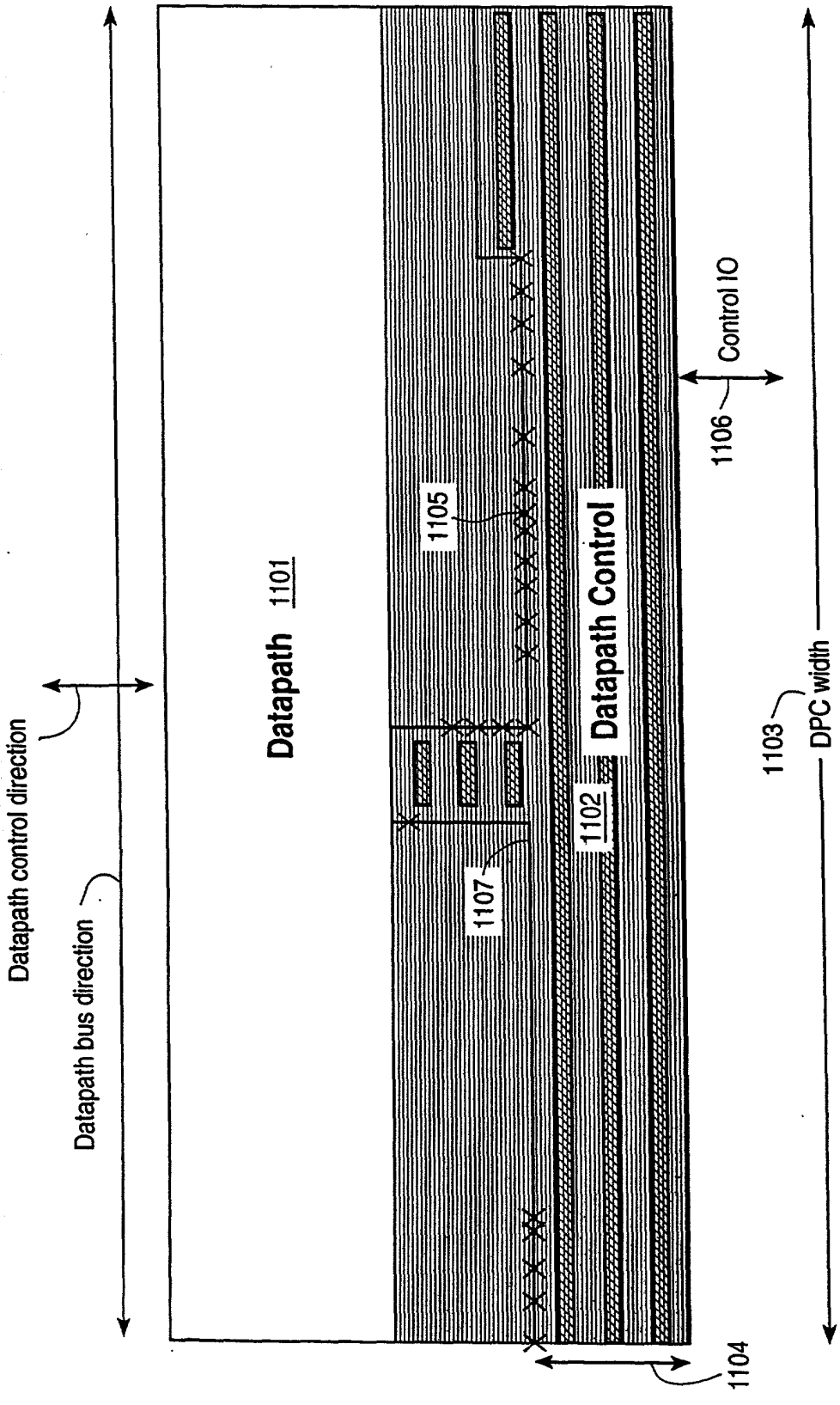

Datapath control direction
Datapath bus direction
Datapath 1101
1105
Datapath Control
1102
1107
Control IO
1106
1103
DPC width
1104
X = Datapath control signal terminals

FIG. 11

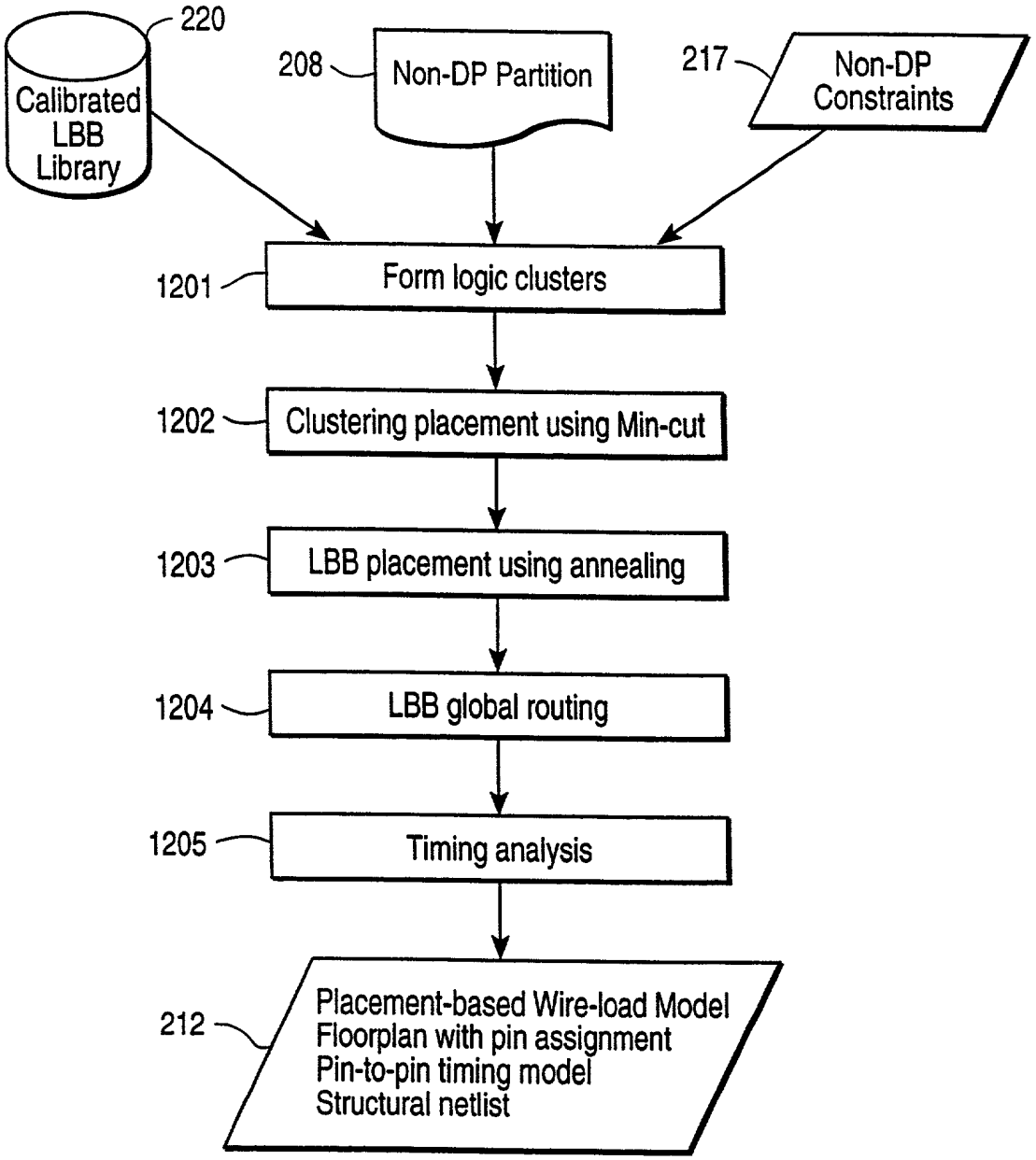

220
Calibrated LBB Library
208
Non-DP Partition
217
Non-DP Constraints
1201
Form logic clusters
1202
Clustering placement using Min-cut
1203
LBB placement using annealing
1204
LBB global routing
1205
Timing analysis
212
Placement-based Wire-load Model
Floorplan with pin assignment
Pin-to-pin timing model
Structural netlist

FIG. 12

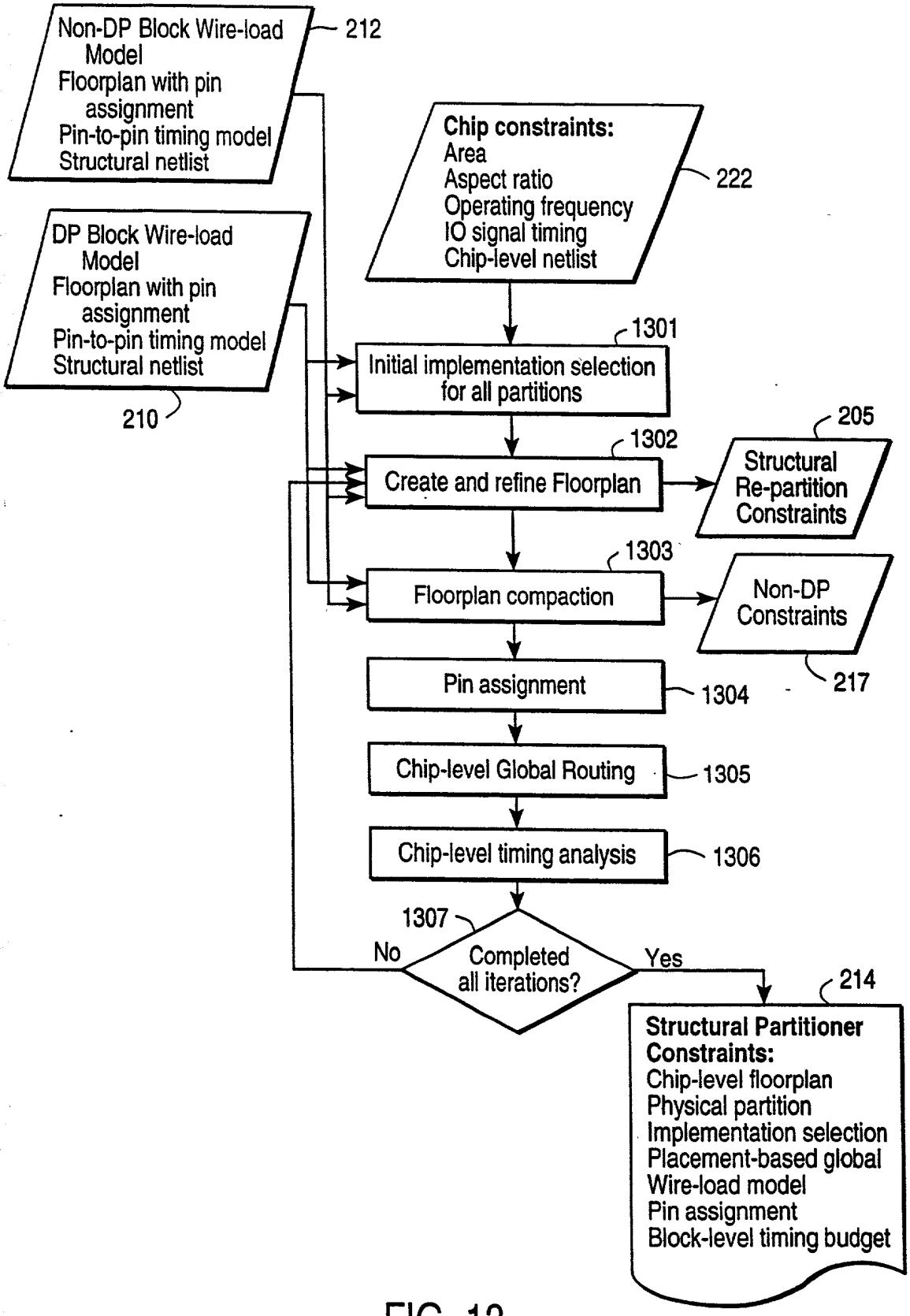

Non-DP Block Wire-load Model
Floorplan with pin assignment
Pin-to-pin timing model
Structural netlist
212
DP Block Wire-load Model
Floorplan with pin assignment
Pin-to-pin timing model
Structural netlist
210
Chip constraints:
Area
Aspect ratio
Operating frequency
IO signal timing
Chip-level netlist
222
1301
Initial implementation selection for all partitions
1302
Create and refine Floorplan
205
Structural Re-partition Constraints
1303
Floorplan compaction
Non-DP Constraints
217
Pin assignment
1304
Chip-level Global Routing
1305
Chip-level timing analysis
1306
1307
Completed all iterations?
No
Yes
214
Structural Partitioner Constraints:
Chip-level floorplan
Physical partition
Implementation selection
Placement-based global
Wire-load model
Pin assignment
Block-level timing budget

FIG. 13

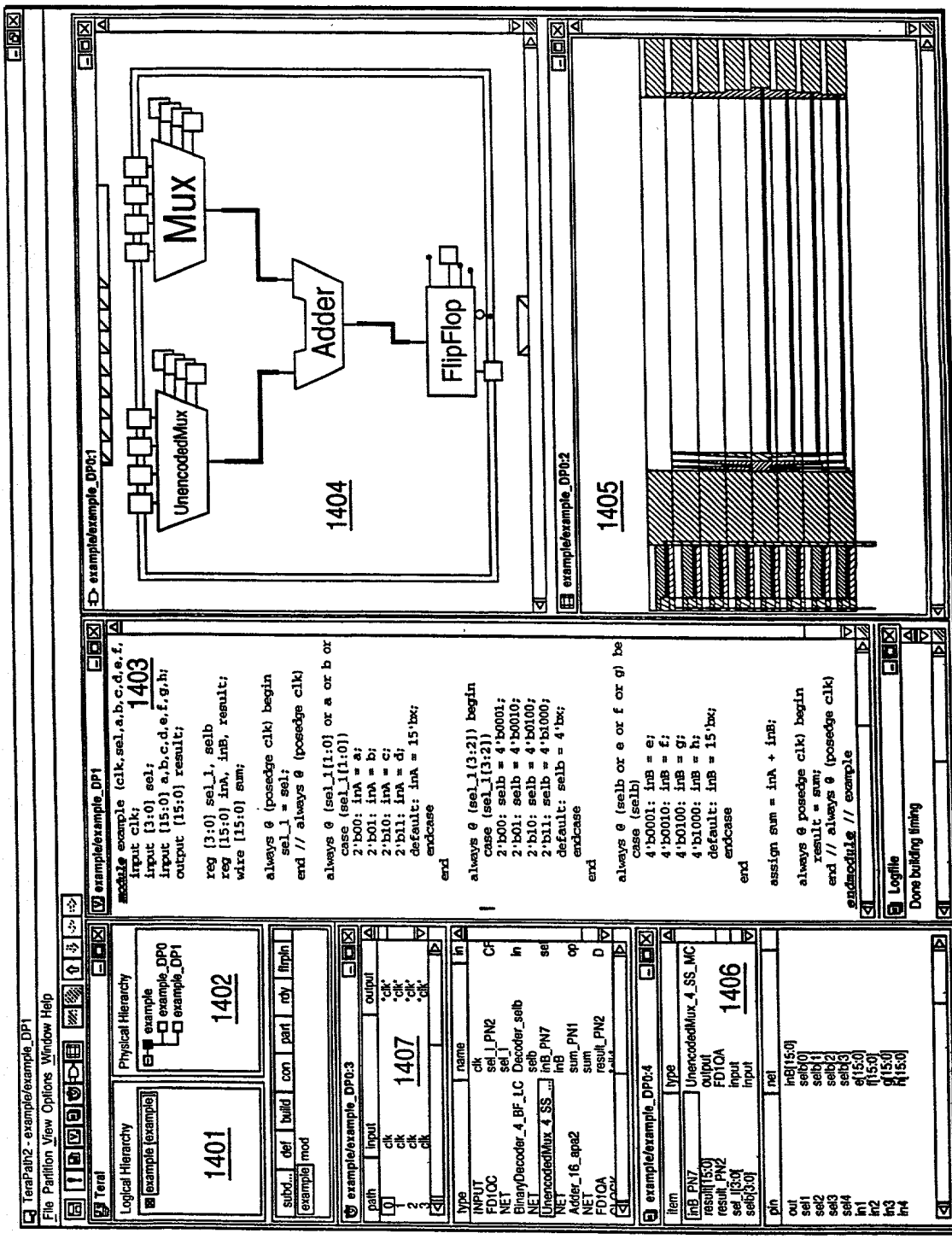

1401
1402
1403
1404
1405
1406
1407
Mux
Adder
FlipFlop
UnencodedMux

FIG. 14